Amendment to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

1.   (Canceled)

2.   (Canceled)

3.   (Currently Amended) The method of claim ~~11~~ 11, further comprising determining an address of ~~an~~ a first extreme value based on a value in a pointer register and based on a correction factor to compensate for one or more errors.

4.   (Canceled)

5.   (Currently Amended) The method of claim ~~11~~ 11, wherein M = 2 and N is greater than two.

6.   (Currently Amended) The method of claim ~~11~~ 11, wherein executing the first machine instruction further includes:
    storing the ~~current~~ M current extreme values in M accumulators; and
    copying the M data elements to the accumulators based on said comparing.

2

7.   (Previously Presented) The method of claim 5, wherein said concurrently comparing the M data elements comprises processing a first data element with a first execution unit of a pipelined processor and processing a second data element with a second execution unit of the pipelined processor.

8.   (Previously presented) The method of claim 5, wherein concurrently comparing the M data elements comprises concurrently processing a first data element and a second data element within a single execution unit of a pipelined processor.

9.   (Currently Amended) The method of claim $\frac{1}{11}$, wherein said concurrently comparing the M data elements to M corresponding current extreme values includes determining whether each of the data elements is less than the corresponding current extreme value.

10.   (Currently Amended) The method of claim $\frac{1}{11}$, wherein said concurrently comparing the M data elements to M corresponding current extreme values includes determining whether each of the data elements is greater than the corresponding current extreme value.

3

11.   (Currently Amended) A method for searching an array of N data elements for an extreme value, the method comprising:

issuing N/M machine instructions to a processor, wherein the processor is adapted to process M data elements in parallel;

executing ~~each~~ a first machine instruction by:

concurrently comparing M data elements to corresponding M current extreme values,

retrieving another M elements in a single fetch cycle to be compared when executing a subsequent machine instruction;

updating accumulators and pointers associated with the M current extreme values based on said comparing, the pointers including one or more pointer registers to store information indicative of addresses of extreme values in the array of N data elements; and

analyzing results of the machine instructions to identify at least a value and a position of at least one extreme value in the array, wherein the at least one extreme value comprises an extreme value occurring more than once in the array, and wherein the position of the at least one extreme value in the array comprises a position of a predetermined one of a first occurrence and a last occurrence of the extreme value occurring more than once in the array.

4

12. (Previously Presented) The method of claim 11, further comprising:

setting up registers for said accumulators and pointers.

13. (Currently Amended) A method comprising:

retrieving a pair of data elements from an array of elements in a single fetch operation, wherein the pair of data elements includes an even data element and an odd data element;

substantially comparing the even element of the pair with an even extreme value;

if the even element of the pair exceeds the even extreme value, storing the even element of the pair as the even extreme value and storing a parameter indicative of an address of the even element of the pair in a pointer register, wherein storing the parameter indicative of an address of the even element of the pair in a pointer register comprises maintaining a first pointer register to store an address for the extreme value of the even data elements;

concurrent with said comparing the even element of the pair with the even extreme value, comparing the odd element of the pair with an odd extreme value;

if the odd element of the pair exceeds the odd extreme value, storing the odd element of the pair as the odd extreme value, and further comprising maintaining a second pointer

5

register to store an address for the extreme value of the odd

data elements; ~~and~~

substantially fetching and comparing remaining pairs of

data elements of the array until all of the data elements of the

array have been processed; and

adjusting at least one of the pointer registers after

processing all of the pairs of data elements to account for a

number of stages in a pipeline.

14.   (Previously Presented) The method of claim 13, further

comprises setting the even extreme value as a function of the

even element of the element pair and setting the odd extreme

value as a function of the odd element of the element pair.

15.   (Previously Presented) The method of claim 13, further

comprises maintaining a first accumulator to store a minimum

value for the even elements and a second accumulator to store a

minimum value for the odd elements.

16.   (Canceled)

17.   (Canceled)

18.   (Previously Presented) The method of claim 13, wherein the method is invoked by issuing N/M machine instructions to a programmable processor, wherein N equals a number of elements in the array, and M equals a number of data elements that the processor can concurrently compare.

19.   (Currently Amended) An apparatus comprising:

a execution pipeline adapted to process M data elements in parallel; and

a control unit adapted to direct the execution pipeline to search an array of N data elements for an extreme value in response to N/M machine instructions, the execution pipeline being configured to:

retrieve M data elements from the array of N data elements in a single fetch cycle;

concurrently compare the retrieved M data elements to corresponding M current extreme values, ~~and~~

update accumulators and pointers associated with the M current extreme values based on said comparing, the pointers including one or more pointer registers to store <u>information indicative of</u> addresses of extreme values in the array of N data elements<u>; and</u>

<u>analyze results of the N/M machine instructions to identify at least a value of at least one extreme value in</u>

7

the array, wherein the at least one extreme value comprises an extreme value occurring more than once in the array, and wherein the position of the at least one extreme value in the array comprises a position of a predetermined one of a first occurrence and a last occurrence of the extreme value occurring more than once in the array.

20. (Previously Presented) The apparatus of claim 19, wherein in response to the machine instructions, the control unit directs the pipeline to set up registers for accumulators and pointers.

21. (Previously Presented) The apparatus of claim 19, wherein the pipeline includes M registers adapted to store accumulators and pointers associated with the extreme values.

22. (Previously Presented) The apparatus of claim 21, wherein the registers include first and second pointer registers to store information indicative of addresses of first and second extreme values of the array.

23. (Original) The apparatus of claim 21, wherein the registers are general-purpose data registers.

8

24. (Previously Presented) The apparatus of claim 19, wherein the pipeline includes M accumulators to store M current extreme values.

25. (Previously Presented) The apparatus of claim 19, wherein the pipeline includes M general-purpose registers to store M current extreme values.

26. (Currently Amended) An article comprising a medium having computer-executable instructions stored thereon for compiling a software program, wherein the computer-executable instructions are adapted to generate N/M machine instructions to search an array of N data elements to find an extreme value, each machine instruction causing a programmable processor to:

retrieve M data elements from an array of N elements in a single fetch operation;

concurrently compare the retrieved M data elements to M ~~a~~ corresponding current extreme values; ~~and~~

update accumulators and pointers associated with the M current extreme values based on said comparing, the pointers including one or more pointer registers to store <u>information indicative of</u> addresses of extreme values in the array of N elements; <u>and</u>

9

analyze results of the N/M machine instructions to identify

at least a value of at least one extreme value in the array,

wherein the at least one extreme value comprises an extreme

value occurring more than once in the array, and wherein the

position of the at least one extreme value in the array

comprises a position of a predetermined one of a first

occurrence and a last occurrence of the extreme value occurring

more than once in the array.

   27.   (Canceled)

   28.   (Original) The article of claim 26, wherein each

machine instruction causes the processor to concurrently process

a first data element and a second data element within a single

execution unit of a pipelined processor.

   29.   (Currently Amended) A system comprising:

a memory device; and

     a processor coupled to the memory device, wherein the

processor includes a pipeline configured to process M data

elements in parallel and a control unit configured to direct the

pipeline to search an array of N data elements for an extreme

value in response to N/M machine instructions, wherein in

10

response to each of the machine instructions, the pipeline being
configured to:

    retrieve M data elements from the array of N data
elements in a single fetch cycle;

    concurrently compare the retrieved M data elements to
corresponding M current extreme values, and

    update accumulators and pointers associated with the M
current extreme values based on said comparing, the
pointers including one or more pointer registers to store
information indicative of addresses of extreme values in
the array of N data elements; and

    analyze results of the N/M machine instructions to
identify at least a value of at least one extreme value in
the array, wherein the at least one extreme value comprises
an extreme value occurring more than once in the array, and
wherein the position of the at least one extreme value in
the array comprises a position of a predetermined one of a
first occurrence and a last occurrence of the extreme value
occurring more than once in the array.

30.   (Canceled)

11

31.   (Previously Presented) The system of claim 29, wherein
the pipeline includes M registers configured to store the
accumulators and pointers.

32.   (Previously Presented) The system of claim 31, wherein
the registers include first and second pointer registers to
store information indicative of addresses of first and second
extreme values of the array.

33.   (Original) The system of claim 31, wherein the
registers are general-purpose data registers.

34.   (Original) The system of claim 29, wherein the memory
device comprises static random access memory.

35.   (Original) The system of claim 29, wherein the memory
device comprises FLASH memory.

36.   (Previously Presented) The method of claim 11,
wherein the ~~at least a value and a position of at least one~~
~~extreme value in the array comprises a value and a position of a~~
~~first occurrence of a minimum value in the array~~ extreme value
occurring more than once in the array is a minimum value, and
wherein the position of the at least one extreme value in the

12

array comprises a position of a first occurrence of the minimum value in the array.

37. (Currently Amended) The method of claim 11, wherein the ~~at least a value and a position of at least one extreme value in the array comprises a value and a position of a last occurrence of a minimum value in the array~~extreme value occurring more than once in the array is a minimum value, and wherein the position of the at least one extreme value in the array comprises a position of a last occurrence of the minimum value in the array.

38. (Currently Amended) The method of claim 11, wherein the ~~at least a value and a position of at least one extreme value in the array comprises a value and a position of a last occurrence of a maximum value in the array~~.

39. (Previously Presented) The method of claim 11, wherein the ~~at least a value and a position of at least one extreme value in the array comprises a value and a position of a first occurrence of a maximum value in the array~~extreme value occurring more than once in the array is a maximum value, and wherein the position of the at least one extreme value in the

13

array comprises a position of a first occurrence of the maximum value in the array.

14